LVC20-216

TensorFlow and PyTorch on Arm Servers

Linaro Virtual Connect 2020

Ashok Bhat, Sr Product Manager

Sep 2020

Agenda

TensorFlow and PyTorch on Arm Servers for on-CPU inference

- What is the end goal?
- What is available today for end users?
- What is being worked on to improve usability and performance?
- How to get involved?

Not being covered

- Training use-case
- Machine learning using Arm + GPU
- Benchmarks and performance comparison

On-CPU Machine Learning (Inference)

Goal: Easy to use, best-in-class performance, ML inference solution on Arm servers using ML specific CPU features

Easy to use	Wide variety of inference workloads	Using Arm architecture features	On latest Arm based hardware
Container images and Python Packages	Image classification	Large core count	INT8, Bfloat16, FP32	Arm Neoverse N1, V1 (Zeus), N2 (Perseus)
Popular ML frameworks support Arm as a first-class citizen	Object detection	Matrix Multiplier Extension	Fujitsu A64FX
	…	SVE/2	…
		…	…

arm

arm

AArch64 packages and images

For machine learning users on Arm servers

Ready-to-use Python Packages

Goal: Readily available TensorFlow and PyTorch packages from standard repositories

Current status

- TensorFlow 1.15 and 2.3 package snapshots available
  - https://snapshots.linaro.org/hpc/python/tensorflow/latest/
- PyTorch nightly package snapshots available
  - https://snapshots.linaro.org/hpc/python/pytorch/latest

Next steps

- Snapshots – Add support for PyTorch v1.6
- Linaro hosted packages – Host ready-to-use packages for TensorFlow and PyTorch
- Work with upstream to provide AArch64 ready packages

Docker recipes

Goal: Recipe to build your own Docker images

Current status

- TensorFlow
  - https://github.com/ARM-software/Tool-Solutions/tree/master/docker/tensorflow-aarch64
  - Versions – 1.15 and 2.3
  - Configurations – Eigen backend, oneDNN(ArmPL), oneDNN(OpenBLAS)
- PyTorch
  - https://github.com/ARM-software/Tool-Solutions/tree/master/docker/pytorch-aarch64
  - Versions – 1.6
  - Configurations – OpenBLAS backend

Next steps

- Upgrade recipes to newer releases
- Add oneDNN(ACL) based configuration

Docker images

Goal: Readily available docker images on par with other architectures

Current status

- Images for Arm Neoverse N1 is available in a staging area
  - TensorFlow 2.3 with Eigen, oneDNN (ArmPL)
  - PyTorch 1.6 with OpenBLAS

Next steps

- Upgrade images to newer releases
- Add oneDNN(ACL) based configuration
- Add images tuned for Fujitsu A64FX
- Provide images on Linaro Docker Hub repo
- Work with upstream to provide images in standard repositories

arm

Best-in-class performance using ML-specific Arm features

Key open source projects

TensorFlow

PyTorch

OpenBLAS

Eigen

oneDNN

Arm Compute Library

Key open source projects for ML on Servers

Frameworks

- ML Framework - TensorFlow
  - Popular open source ML framework
  - Has multiple backends on x86 – Eigen GEBP, oneDNN (with BLAS, direct kernels, JIT)
- ML Framework - PyTorch
  - Popular ML framework
  - Has multiple backends on x86 – NNPACK, OpenBLAS, oneDNN (with BLAS, direct kernels, JIT)

Key open source projects for ML on Servers

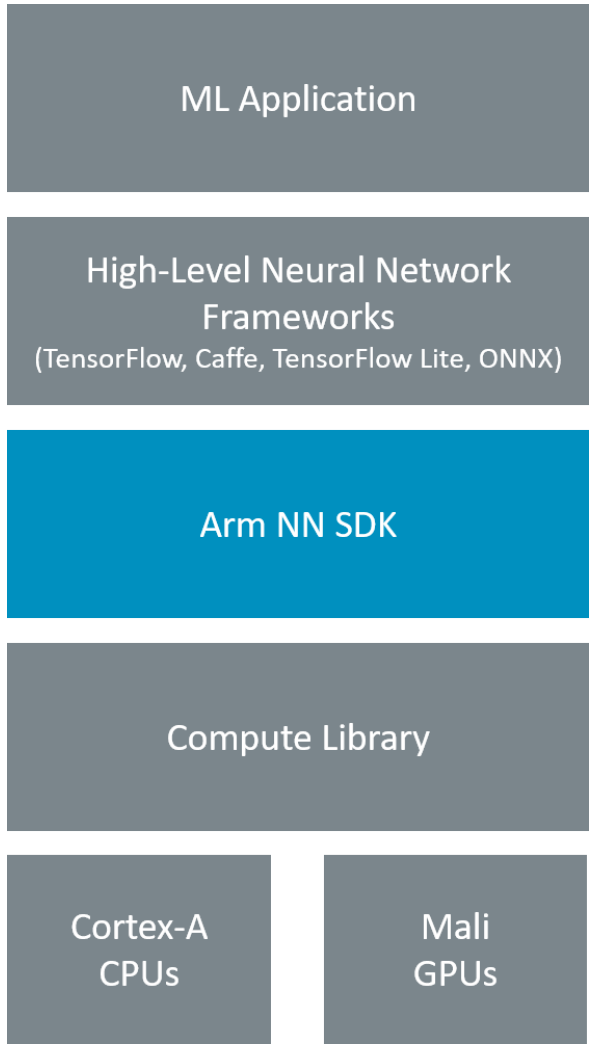
Libraries

- Library - Eigen
  - Eigen is a C++ template library for linear algebra: vectors, matrices, and related algorithms
  - TensorFlow heavily uses Eigen to represent internal data structures and their operations.
  - Eigen's GEBP kernel is used as a default CPU backend for FP32 contraction kernel
- Library - oneDNN
  - Intel's ML acceleration open-source library – Integrated with all major frameworks
  - Experimental support for AArch64
- Library - Arm Compute Library
  - Open source ML acceleration library for Arm used in edge/mobile use-cases
  - Contains high level operators which can be used in oneDNN
- Library - OpenBLAS
  - Most common open source BLAS backend

Arm Compute Library

A software library for computer vision and machine learning

- Collection of low-level functions
  - Optimized for Arm CPU and GPU architectures
  - Targeted at image processing, computer vision, and machine learning.
- Available free of charge under a permissive MIT open source license.
- Used to accelerate ArmNN (Arm's inference engine for CPUs, GPUs and NPUs)

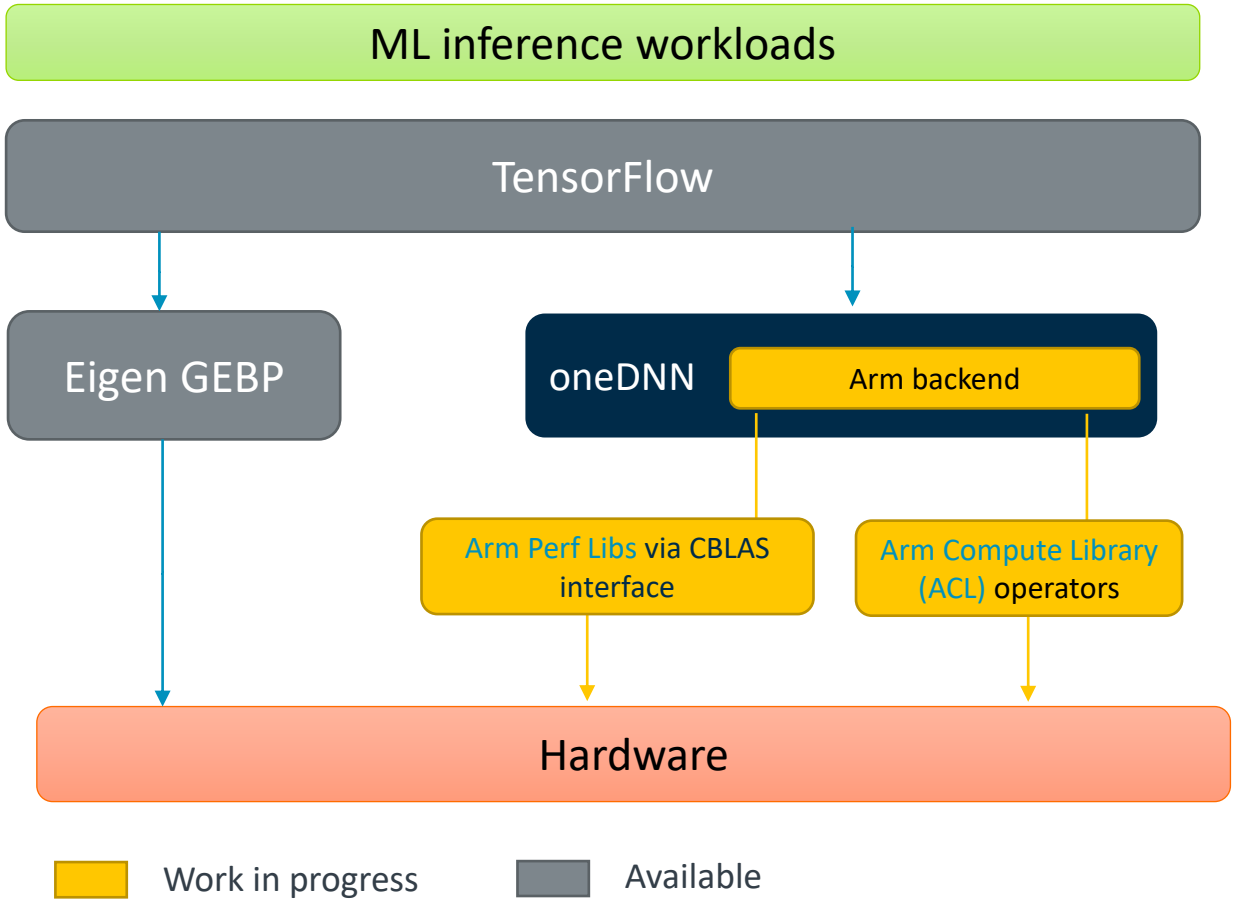


arm

oneDNN primitives – Implementation options

Primitive implementation	Notes	x86_64 libraries	Arm libraries
C++ reference code	Slowest, provided for correctness	NA	NA
GEMM based	Use BLAS library via CBLAS interface	Intel MKLML (a subset of Intel MKL)	Arm Performance Libraries (ArmPL)
Optimized primitives	Typically uses hand-coded intrinsic or assembly	xybak (JIT assembler)	Arm Compute Library

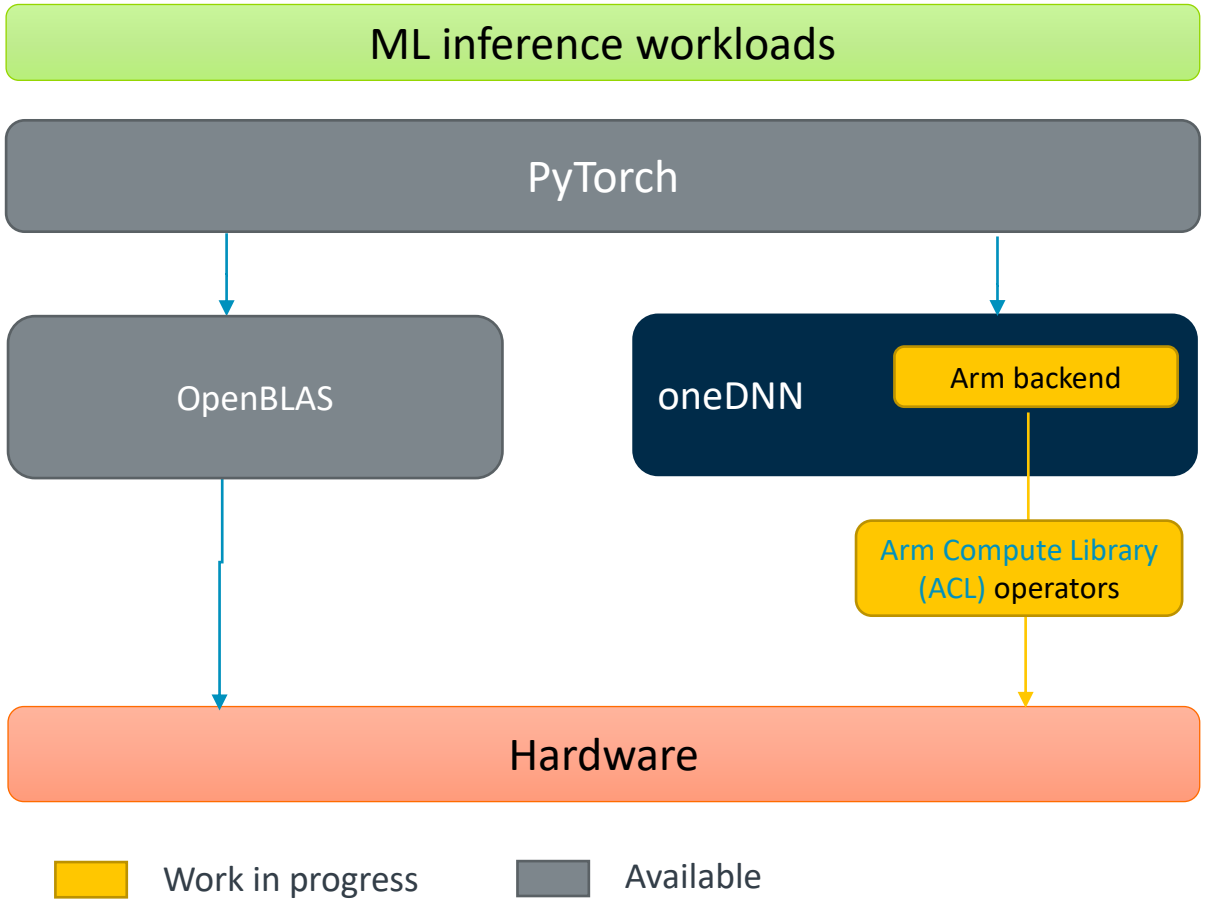
TensorFlow software stack on Arm – Status and Plan



Timeline	Library options
2.2 (May 20)	Eigen (FP32)
2.3 (Jul 20)	Eigen (FP32)
2.4* (Q4, 20)	Eigen (FP32) oneDNN (ArmPL via CBLAS)
2.5* (Q1, 21)	Eigen (+SVE) oneDNN (ArmPL via CBLAS) oneDNN (ACL) (FP32, INT8)

* Future release information (version and date) is Arm's estimate based on previous releases.

PyTorch software stack on Arm – Status and Plan



Timeline	Library options
1.6 (Jul 20)	OpenBLAS (FP32)
1.7* (Oct 20)	OpenBLAS (FP32)
1.8* (Dec 20)	OpenBLAS (FP32) oneDNN (ACL) (FP32, INT8)

* Future release information (version and date) is Arm's estimate based on previous releases.

Data type support

Status and plan

Data type	TF Eigen	TF ACL (via oneDNN)	PyTorch OpenBLAS	PyTorch ACL (via oneDNN)
FP32 type	Yes	Q4 CY20	Yes	Q4CY20
INT8 type		Q4 CY20		Q4 CY20
BF16 type	Q4 CY20	Future		Future

Not planned

arm

Wrap Up

Get involved in Machine Learning on Arm

Try	Try the Docker recipes/images to run TensorFlow and PyTorch on AArch64
Learn	Learn about the Arm Compute Library
Provide Feedback	Provide feedback on performance for your applications on Arm machines
Get involved	Get involved in the open source development of ML inference on Arm Weekly public meeting to get involved at: https://bit.ly/arm-server-ml

arm

Thank You
Danke
Merci
谢谢
ありがとう
Gracias
Kiitos
감사합니다
धन्यवाद
شكرًا
ধন্যবাদ
תודה